IRIA

[CE]NTRE DE ROCQUENCOURT

Institut National
de Recherche
en Informatique
et en Automatique

Domaine de Voluceau
Rocquencourt
B.P.105
78153 Le Chesnay Cedex
France
Tél: 954 90 20

Rapports de Recherche

N° 103

A TWO-STAGE FEASIBLE DIRECTION ALGORITHM FOR NONLINEARLY CONSTRAINED OPTIMIZATION

José HERSKOVITS

Novembre 1981

A TWO-STAGE FEASIBLE DIRECTION ALGORITHM FOR NONLINEARLY CONSTRAINED OPTIMIZATION

José HERSKOVITS *

Abstract :

We present a feasible direction algorithm, based on lagrangian concepts, for the solution of the nonlinear programming problem with equality and inequality constraints. At each iteration a descent direction is defined , by modifying it, a feasible and descent direction is obtained. The linear search procedure assures the global convergence of the method, and the feasibility of all the iterates.

We prove the global convergence of the algorithm, and show the results obtained in the resolution of some test problems. Although the present version of the algorithm does not include any second order approximation, like quasi-Newton methods, these numerical results exhibit a behaviour comparable of that of the best methods known at present for nonlinear programming.

Résumé :

On présente un algorithme de directions réalisables utilisant le lagrangien, pour résoudre les problèmes de programmation non linéaires avec contraintes d'égalité et d'inégalité. A chaque itération, une direction de descente est d'abord calculée ; elle est ensuite modifiée pour obtenir une direction de descente réalisable ; enfin une recherche linéaire est effectuée pour assurer la réalisabilité de l'itéré suivant, et la convergence du processus.

On démontre la convergence globale de la méthode, et on présente les résultats numériques obtenus sur quelques problèmes de la littérature.

Bien que la présente méthode ne contienne aucune approximation du second ordre -comme dans les méthodes de quasi-Newton- ces résultats exhibent un comportement analogue à celui des meilleures méthodes actuellement connues en programmation non linéaire.

This research was supported in part by the Brazilian Research Conseil (CNPq).

* Permanent address : COPPE/UFRJ, Caixa Postal 68503-ZC-00,21944 RIO DE JANEIRO, RJ, Brazil.

1. INTRODUCTION AND PRELIMINARIES

The general non linear constrained optimization problem can be defined as follows :

$$\min_{x} f(x) \tag{1.1}$$

$$\begin{aligned} &\text{subject to} \quad g_i(x) \le 0 \quad ; \quad i = 1,\dots,m \\ &\text{and} \qquad\quad\ g_i(x) = 0 \quad ; \quad i = m+1,\dots,m+p \end{aligned}$$

where $f(x)$ and $g_i(x)$ denote real valued functions of a vector x in the n-dimensional Euclidian space R^n.

A considerable research effort has been done to obtain efficient and reliable methods for the solution of this problem. Without trying to make a survey of this area, we can mention different approaches concerning each of the components of the problem. That is, minimizing the function, "solving" the equality constraints and verifying the inequalities. A very interesting survey in this sense, has been done by Fletcher [9]. The best known methods of unconstrained optimization are concerned with the minimization of the function. We shall mention steepest-descent, quasi-Newton, and conjugate gradient methods, when only first derivative information is considered.

Equality constraints may be eliminated, linearized, or penalized. Methods using simple penalty functions are robust, but when a good precision is required they may give ill-conditioning.

Augmented Lagrangians [3,21] avoid in general the ill-conditioning, but they are less robust and their precision is not very good.

When the constraints are linearized, the first idea is to project the steepest descent or the quasi-Newton direction on the tangent subspace. In that case we have the projected gradient method, or the reduced gradient one [22,1,2]. When the constraints are not linear, they need some feasibility restoring scheme. It is also possible to combine the projected gradient direction with a feasibility improvement step [10].

The combination of a linear approximation of the constraints with a quadratic approximation of the objective function, is the basis of a family of methods which solve a quadratic programming subproblem in each iteration. They give a direction tangent to the active equality constraints and improve feasibility automatically.

It can be proved that if all the constraints are active, directions given by projected gradient, augmented Lagrangian and quadratic programming subproblem methods, are similar.

Inequality constraints may be treated as equalities, if the set of active constraints at the optimum is known. In practice, it is very difficult to make a good prediction of the active set and the methods that use this approach are subject to the so-called "jamming" problem.

Barrier and simple penalty methods search automatically the active inequality constraints but they may give ill-conditioning, and inexact penalty functions produce no feasible points.

The quadratic programming subproblem methods are naturally extended to inequality constraints. They identify very efficiently the active set [21,11,24], but the feasible region of the subproblem may be empty or unbounded.

Duality is at the origin of a family of approaches in the treatment of equality and inequality constraints. Minimax and augmented Lagrangian methods are natural applications of this theory.

If some proper update rule for the Lagrange multipliers is stated, the exact minimization of the intermediate unconstrained problem can be replaced by a single minimization step. This idea is the starting-point of Tapia's diagonalized multipliers methods [24] and Biggs' recursive quadratic programming [4,5,6]. These methods need less calculations than quadratic programming subproblem approaches, but some active set strategy must be adopted.

We shall remark that, in general, the final convergence qualities of a constrained optimization method cannot be better than those of the unconstrained minimization technique included in it.

In this work, we present a strong and efficient method for the solution of problem 1.1, with good local and global convergence qualities.

This is obtained by establishing an update rule for the Lagrange multipliers, without employing active set strategies. Inequality constrained subproblems and ill-conditioning given by penalty functions are also avoided. The method gives a feasible direction for the inequality constraints and, in consequence, feasible intermediate points. A superlinear local convergence may be obtained by including an approximation of the Hessian of the Lagrangian.

An algorithm for inequality constrained problems is given in Section 2, and its global convergence is proved in Section 3. The equality constraints are introduced in Section 4, and some numerical examples are considered in Section 5.

NOTATIONS

All vector spaces are finite dimensional, the space of all $n \times m$ matrices is denoted by $R^{n\times m}$ and the transpose of M by M^T. If ϕ is a real valued function in R^n, then

$$\nabla\phi(x) \equiv \left(\frac{\partial\phi(x)}{\partial x_1}, \frac{\partial\phi(x)}{\partial x_2}, \ldots \frac{\partial\phi(x)}{\partial x_n}\right)^T .$$

We call Ω the feasible region for the inequality constraints, that is

$$\Omega \equiv \{x \in R^n; g_i \leq 0 \quad , \quad i = 1,\ldots,m\} ,$$

and denote by

$$g(x) \equiv [g_1(x), g_2(x), \ldots, g_{m+p}(x)]^T$$

and

$$A(x) \equiv [\nabla_1 g(x), \nabla g_2(x), \ldots, \nabla g_{m+p}(x)] .$$

DEFINITIONS

Consider the following definitions :

Definition 1.1 : A point $\bar{x}$ is a "stationary point" of problem (1.1) if there exists a vector $\bar{\lambda}$ in R^m such that the following requirements are simultaneously satisfied :

$$g_i(\bar{x}) \leq 0 \quad ; \quad i = 1,\ldots,m$$

$$g_i(\bar{x}) = 0 \quad ; \quad i = m+1,\ldots,m+p$$

$$\bar{\lambda}_i g_i(\bar{x}) = 0 \quad ; \quad i = 1,\ldots,m$$

and

$$\nabla f(x) + \sum_{i=1,m+p} \bar{\lambda}_i \nabla g_i(\bar{x}) = 0 .$$

Definition 1.2 : A "Kuhn-Tucker point" of the problem (1.1) is a stationary point associated to a vector $\bar{\lambda}$ verifying

$$\bar{\lambda}_i \geq 0 \quad ; \quad i = 1,\ldots,m$$

Defintion 1.3 : $d \in R^n$ is a "descent direction" of a real continuously differentiable function $\phi(x)$ in R^n, if

$$d^T \nabla\phi(x) < 0 .$$

Definition 1.4 : $d \in R^n$ is a "feasible direction" [14] of problem (1.1) at $x \in \Omega$ if for some $\tau > 0$ we have

$$x + td \in \Omega \quad \text{for all } t \in [0,\tau].$$

Defintion 1.5 : A point $\bar{x}$ is a "regular point" of the problem (1.1) if the elements of the set of vectors composed by all

$$\nabla g_i(\bar{x}) \quad ; \quad i = 1,\dots,m \quad \text{so that} \quad g_i(\bar{x}) = 0$$

and all

$$\nabla g_i(\bar{x}) \quad ; \quad i = m+1,\dots,m+p$$

are linearly independent.

2. STATEMENT OF THE ALGORITHM

In this section, we shall consider the inequality constrained problem

$$\min_{x} f(x) \tag{2.1}$$

subject to $\quad g_i(x) \le 0 \quad ; \quad i = 1,\dots,m$.

The equality constraints will be introduced later.

We suppose that there exists a real number $\underline{a}$ so that the region

$$\Omega_a \equiv \{x \in \Omega \ ; \ f(x) \le a\}$$

is a compact with nonempty interior. We suppose also that f is $\mathbb{C}^1$ and g_i are C^2 in Ω_a, and that all $x \in \Omega_a$ are regular points of the problem.

We shall define a feasible direction algorithm based on Lagrangian concepts, for the solution of problem (2.1). The method constructs a sequence $\{x^k\}$, starting from an initial strictly feasible point, verifying

$$g_i(x^k) < 0 \quad ; \quad i = 1,\dots,m \quad \text{and} \quad k = 0,1,2,\dots,$$

and converging to a Kuhn-Tucker point of the problem.

A direction $d \in R^n$ is computed in two stages. First a descent direction $d_0(x^k)$ is defined; by modifying it, the feasible direction $d(x^k)$ is obtained without loosing the descent quality of $d_0(x^k)$.

A linear search is stated, in order to guarantee the global convergence of the method and the strict feasibility of all the iterates.

In the definition of the first step direction $d_0(x^k)$, we shall apply a scheme previously given by Zouain and the author in a non-linear programming method for structural optimization problems [13]. An important feature, is that all the constraints are considered in each iteration, and it is not necessary to perform any active set strategy.

The algorithm for the solution of problem (2.1) is stated as follows :

Let $\rho_0 > 0$, $\alpha \in (0,1)$, $\gamma_0 \in (0,1)$ and $r_i(x) > 0$;

$i = 1,\ldots,m$ continuous in Ω_a.

Step 0 : Select a strictly feasible initial point $x^0 \in \Omega_a$, and the values of α, γ_0, and ρ_0. Set $\rho = \rho_0$.

Step 1 : Compute $\lambda_0 \in R^m$ and $d_0 \in R^n$ by solving the linear system of equations

$$d_0 = -[\nabla f(x) + \sum_{i=1}^{m} \lambda_{0i} \nabla g_i(x)] \tag{2.2}$$

$$d_0^T \nabla g_i(x) = -r_i(x)\lambda_{0i} g_i(x) \quad ; \quad i = 1,\ldots,m \tag{2.3}$$

If $d_0 = 0$, stop.

Step 2 : Compute $\rho_1 = \dfrac{1-\alpha}{\sum_{i=1}^{m} \lambda_{0i}}$ (2.4)

If $0 < \rho_1 < \rho$, set

$$\rho = \frac{1}{2}\rho_1$$

Compute $\lambda \in R^m$ and $d \in R^n$ by solving the linear system of equations

$$d = -[\nabla f(x) + \sum_{i=1}^{m} \lambda_i \nabla g_i(x)] \tag{2.5}$$

$$d^T \nabla g_i(x) = -[r_i(x)\ \lambda_i\ \ g_i(x) + \rho\ |d_0|^2] \quad ; \quad i = 1,\ldots,m \tag{2.6}$$

Step 3 : For each constraint, set $\gamma_i = \gamma_0$ if $\lambda_i \geq 0$, $\gamma_i = 1$ if $\lambda_i < 0$. Call τ the smallest τ_i such that

$$g_i(x+td) \leq \gamma_i\ g_i(x)$$

for $t \in (0,\tau_i]$ and $i = 1,\ldots,m$.

Find $\bar{t}$ so that

$$f(x + \bar{t}d) = \min f(x+td)$$

over $t \in [0,\tau]$

Step 4 : Set the new iterate

$$\bar{x} = x + \bar{t}d \tag{2.9}$$

Step 5 : Go to step 1.

Denoting by R and G the $m \times m$ diagonal matrices with $R_{ii}(x) = r_i(x)$ and $G_{ii}(x) = g_i(x)$, then (2.2) and (2.3) are equivalent to

$$d_0 = -(\nabla f + A\ \lambda_0) \tag{2.10}$$

and

$$A^T d_0 = -R\ G\ \lambda_0\ , \tag{2.11}$$

where $\nabla f, G$ and A are computed at x.

Substituting d_0 in (2.10) into (2.11) we get an expression for λ_0

$$\lambda_0 = -(A^T A - RG)^{-1} A^T \nabla f \tag{2.12}$$

and consequently

$$d_0 = -[I - A(A^T A - RG)^{-1} A^T]\nabla f \tag{2.13}$$

In a similar way, we get

$$\lambda = (A^T A - RG)^{-1} (-A^T \nabla f + \rho |d_0|^2 e) \tag{2.14}$$

and

$$d = d_0 - \rho |d_0|^2 A(A^T A - RG)^{-1} e \tag{2.15}$$

where

$$e \equiv \underbrace{(1,1,\ldots,1)}_{m}^T .$$

In consequence, λ and d are given explicitely by (2.14) and (2.15).

Before carrying out the mathematical development leading to the proof of global convergence, we shall make some remarks explaining the behaviour of the algorithm and the ideas behind its construction.

- At the stationary points of problem (1.2), λ and λ_0 are equal to the Lagrange multipliers, according to definition 1.1. At these points, d and d_0 are zero.

- d_0 is the steepest descent direction for the function

$$L(x) = f(x) + \sum_{i=1,m} \lambda_{0i}\, g_i(x)$$

- Equalities (2.3) can be considered as an updating rule for λ_0. They force d_0 to point to the constraints associated to a positive λ_{0i}-parameter.

  This fact, which performs an automatic selection of the active set of constraints, is confirmed in the statement of the linear search scheme.

  A condition equivalent to (2.3), is verified in an algorithm for structural optimization problems developped by Segenreich, Zouain and the author [23] .

- If the minimum of problem (2.1) is an interior point, d_0 approaches the steepest descent direction.

- d_0 is contained in the subspace tangent to the active constraints. Then, when there are active constraints, d_0 may point towards the exterior of the feasible region.

- Comparing (2.13) with expression (11.25) in ref. [16], we can see that d_0 becomes the projected gradient method when all the constraints are active.

- In order to get a feasible direction when there are active constraints, the tangent direction d_0 is modified to obtain a secant direction d. This is done by adding a positive element in the right hand side of (2.3), getting (2.6).

- Strict feasibility condition is needed to avoid stationary which are not Kuhn-Tucker points.

- The stepsize procedure maintains a monotone decrease of the function and acts as a barrier, in order to escape from the constraints with negative λ. It also guarantees strict feasibility at each iteration.

3. GLOBAL CONVERGENCE

We must show that the algorithm given in section 2 is correctly stated, in other words, that it is possible to execute all the steps defined above.

The global convergence will be proved by showing successively that :

i) d_0 is a descent direction of f .

ii) d is a descent direction of f , and also a feasible direction of the problem (2.1).

iii) The algorithm produces a sequence of points converging to a stationary point of the problem.

iv) If a stationary point is a point of convergence of the algorithm, it must verify Kuhn-Tucker conditions.

Lemma 3.1. The matrix $W = (A^TA - RG)$, computed at $x \in \Omega_a$, is positive definite.

Proof. Consider a vector $y \in R^m$; since $r_i\, g_i \le 0$, we deduce

$$y^T(A^TA - RG)y \ge 0 \ .$$

Suppose now $y^T(A^TA - RG)y = 0$,

then $\sum_{i=1}^{m} r_i\, g_i\, y_i^2 = y^TA^TAy$

We deduce $y_i = 0$

for those i such $g_i < 0$,

and also $Ay = 0$

or $\sum_{i=1}^{m} y_i\, \nabla g_i = 0$.

Then, by the regularity conditions

$$y = 0$$

and W is positive definite. □

Lemma 3.2. Consider the $n \times n$ matrix

$$Z = A(A^TA - RG)^{-1}\, A^T$$

computed at $x \in \Omega_a$, and let u_Z be an eigenvalue of Z. Then ,

$$0 \le u_Z \le 1 .$$

Proof. If x is the eigenvector corresponding to u_Z, we can write

$$Zx = u_Z x$$

or
$$A(A^TA - RG)^{-1}A^Tx = u_Z x$$

When $A^Tx = 0$, u_Z is also zero and the lemma is true. If $A^Tx \neq 0$, call

$$y = A^Tx .$$

Then
$$A^TA(A^TA - RG)^{-1}y = u_Z y$$

or
$$[I + RG(A^TA - RG)^{-1}]y = u_Z y$$

Premultiplying both sides by $y^T(A^TA - RG)^{-1}$, we have

$$y^T(A^TA - RG)^{-1}RG\ (A^TA - RG)^{-1}y = (u_{Zi}-1)y^T(A^TA - RG)^{-1}y ,$$

whose left side is negative or zero. Considering lemma (3.1) we deduce

$$u_Z - 1 \le 0 ,$$

and as Z is positive semidefinite, the lemma is proved. □

Lemma 3.3. The vector d_0, defined in step 1 of the algorithm, is a descent direction of f.

Proof. Premultiplying both sides of (2.13) by ∇f^T, we get

$$\nabla f^T d_0 = - \nabla f^T[I-Z]\nabla f \quad , \qquad (3.1)$$

and it follows that

$$\nabla f^T d_0 \leq 0 .$$

Let us consider the case

$$\nabla f^T d_0 = 0 .$$

Then, (3.1) is equivalent to

$$\nabla f^T \nabla f = \nabla f^T Z \nabla f ,$$

and, considering lemma 3.2, we deduce that ∇f is an eigenvector of Z and $u_Z = 1$ the corresponding eigenvalue. That is

$$Z \nabla f = \nabla f ,$$

and substituting in (2.13), we get

$$d_0 = 0.$$

In consequence, d_0 verifies definition 1.3. □

<u>Lemma 3.4</u>. The direction $d(x^k)$, defined in Step 2 of the algorithm, is a descent direction of f.

<u>Proof</u>. In consequence of expression (2.15) for d, we have

$$\nabla f^T d = \nabla f^T d_0 - \rho \, |d_0|^2 \, \nabla f^T A(A^T A - RG)^{-1} e .$$

It follows from (2.12) that

$$\nabla f^T d = \nabla f^T d_0 + \rho \, |d_0|^2 \, \lambda_0^T e. \tag{3.2}$$

If $\lambda_0^T e \leq 0$,

we get $\nabla f^T d \leq \nabla f^T d_0$.

Then, d is a descent direction of f.

Suppose now that

$$\lambda_0^T e > 0 ;$$

considering condition (2.4) for ρ and (3.2), we obtain

$$\nabla f^T d \le \nabla f^T d_0 + (1-\alpha) |d_0|^2 \quad . \tag{3.3}$$

From expression (2.13) for d_0, we have

$$\nabla f^T d_0 = - \nabla f^T [I - Z] \nabla f$$

and $$d_0^T d_0 = \nabla f^T [I - Z][I - Z] \nabla f \ .$$

It follows from Lemma 3.2, that [I - Z] is semi-positive definite and the eigenvalues are less or equal than one, then

$$|d_0|^2 \le - \nabla f^T d_0 \ . \tag{3.4}$$

Substituting in (3.3), we get

$$\nabla f^T d \le \alpha \nabla f^T d_0 \ , \tag{3.5}$$

which implies that d is a descent direction of f. □

Note that (3.5) is also valid when $\lambda_0^T e \le 0$. This inequality is important because it gives an upperbound on the directional derivative of f in the directio d as a function of the derivative in the direction d_0.

The linear search procedure, stated in Step 3 of the algorithm, guarantees that

$$x^k \in \Omega_a \quad ; \quad k = 1,2,...$$

if $x^0 \in \Omega_a$. Since Ω_a is compact and f, g_i are continuously differentiable, $|\nabla f|$ and $|\nabla g_i|$ are bounded in Ω_a. We can deduce, as a consequence of lemma 3.1, that W^{-1} is also bounded in Ω_a. Then, it follows from (2.12) that $\lambda_0^T e$ has a positive bound M, and we can write

$$\lambda_0^{kT} e \leq M \quad ; \quad k = 0,1,2,\ldots .$$

In consequence, ρ has a positive lower bound :

$$\rho \geq \min\{\rho_0, (1-\alpha)/2.M\}$$

In order to guarantee the usefulness of d as descent direction, we need the following corollary of lemma 3.4.

<u>Corollary 3.1</u>. There exists a positive constant β such that the condition

$$|d_0|^2 \geq \beta\ |d|^2 \tag{3.6}$$

is verified for all $x \in \Omega_a$.

<u>Proof</u>. If we call

$$d_n = -\ \rho\ |d_0|^2\ A(A^TA-RG)^{-1}\ e$$

we have $\quad d = d_0 + d_n$,

then $\quad |d| \leq |d_0| + |d_n|. \qquad (3.7)$

It follows from inequality (3.4) that

$$|d_0|^2 \leq |\nabla f|\ |d_0|\ ;$$

in consequence

$$|d_n| \leq \rho\ |\nabla f|\ |A(A^TA-RG)^{-1}\ e|\ |d_0\ |.$$

Then, from compacity of Ω_a, there is a positive constant β_1 such that

$$|d_n| \leq \rho\, \beta_1\, |d_0|$$

is verified in Ω_a. It follows from (3.7) that

$$|d| \leq (1 + \rho\, \beta_1)\, |d_0| \quad ,$$

and

$$\beta = \frac{1}{1 + \rho\, \beta_1}$$

verifies the requirements of the corollary. □

Note that (3.6) implies in particular that if d_0 is zero, d is zero also. The contrary is also true; if

$$d = 0 \; ,$$

it follows from lemma 3.4 that

$$d_0^T \;\nabla f \geq 0 \; ,$$

and in consequence of lemma 3.3, it is

$$d_0 = 0 \; .$$

We are now in a position to show that d is a feasible direction of the problem.

Lemma 3.5. There is a real positive number τ^{max} such that d is a feasible direction of the problem (2.1), according to definition 1.4, for all $\tau \in (0, \tau^{max}]$ for all $x \in \Omega_a$.

Proof. For each i, let $u_i < \infty$ be an upperbound of the eigenvalues of $\nabla^2 g_i$ in Ω_a. Using a second order Taylor's development, we get

$$g_i(x+td) \le g_i(x) + t\, d^T \nabla g_i(x) + \frac{1}{2} u_i\, t^2\, |d|^2 .$$

Therefore, if the inequality

$$g_i(x) + t\, d^T \nabla g_i(x) + \frac{1}{2} u_i\, t^2\, |d|^2 \le \gamma_i\, g_i(x)$$

is satisfied, condition (2.7) is satisfied as well.

By (2.6), this is equivalent to

$$g_i(x)\ [(1-\gamma_i) - r_i \lambda_i t] - \rho t\, |d_0|^2 + \frac{1}{2} u_i\, t^2\, |d|^2 \le 0 .$$

It follows from (3.6) that, if

$$g_i(x)\ [(1-\gamma_i) - r_i \lambda_i t] + (\frac{1}{2} u_i\, t^2 - \rho\, \beta\, t)\ |d|^2 \le 0 , \qquad (3.8)$$

then (2.7) is true.

In order to have an interesting geometric interpretation, we shall guarantee (3.8) by imposing

$$\frac{1}{2} u_i\, t^2 - \rho\, \beta\, t \le 0 \qquad (3.9)$$

and

$$(1-\gamma_i) - r_i \lambda_i t \ge 0 , \qquad (3.10)$$

where (3.9) gives a condition on t due to the curvature of the constraints.

If $u_i \le 0$, (3.9) is satisfied for all positive t, even when $\rho = 0$.

Inequation (3.10) is verified for all positive t, when $\lambda \le 0$, even if $\gamma_i = 1$. If $\lambda > 0$, which implies $\gamma_i < 1$, it must be

$$t \le \frac{1 - \gamma_i}{r_i \lambda_i} .$$

Considering all the constraints, in any $x \in \Omega_a$, we deduce that all $\tau \in (0,\tau^{max}]$, with

$$\tau^{max} = \text{min. of positive}\left(\frac{2\rho\beta}{u_i}\ ,\ \frac{1-\gamma_i}{r_i\lambda_i^{max}}\right)$$

verifies (2.7). It follows that d is a feasible direction of the problem (2.1). □

Note that, if $\lambda_i^{max} \leq 0 \quad ; \quad i = 1,\ldots,m$

and $u_i \leq 0 \quad ; \quad i = 1,\ldots,m$

any value of $\tau^{max} > 0$ is useful.

In particular, if all the constraints are linear, it is possible to take $\rho = 0$, which avoids Step 2 of the algorithm.

An important consequence of the last lemma is that d does not lead to zig-zags.

It follows from lemma 3.1 that, for a given $x \in \Omega_a$, λ_0, d_0, λ and d are uniquely determined by expressions (2.12) to (2.15). Because f and g_i are continuously differentiable in Ω_a, we can deduce that λ_0, d_0, λ and d are continuous functions of x, in Ω_a.

Then, the map

$$D(x^k) = d^k$$

which selects a direction d^k in each point x^k is closed. The map

$$M(x^k,d^k) = x^{k+1}$$

corresponding to the constrained minimization in the direction d^k, stated in Step 3 of the algorithm, is also closed in consequence of lemma 3.5. Zigzagging, sometimes present in feasible direction methods, can be explained by not closed algorithmic maps. See Luemberger [16], chapter 11.

The proof of the global convergence of the method will be done in two parts. First, we show the convergence to a stationary point of the problem, and then, that the point of convergence satisfies Kuhn-Tucker conditions.

In the proof of the next lemma, we shall apply a technique similar to that developed by Han in a global convergence theorem in ref. [12].

Lemma 3.6. Any accumulation point $\bar{x}$ of the sequence $\{x^k\}$ generated by the algorithm is a stationary point of the problem.

Proof. If the sequence terminates, $d_0 = 0$ and (2.2) and (2.3) give the result of the lemma.

Suppose that the sequence

$$\{x^k\} \to \bar{x} \ ; \ k \in K \ .$$

Since d is a continuous function of x,

$$d^k \to \bar{d}$$

where $\bar{d} = d(\bar{x})$.

If $d \neq 0$,

we can take $\bar{t}$ so that

$$f(\bar{x} + \bar{t}\bar{d}) = \min_{0 \le t \le \tau} (x + td) \ ,$$

with τ defined in Step 3 of the algorithm.

By lemma 3.5,

$$\tau > 0 \ ,$$

and as $\bar{d}$ is a descent direction of $f(x)$, we have

$$f(\bar{x} + \bar{t}\bar{d}) < f(\bar{x}) .$$

Call

$$\beta = f(\bar{x}) - f(\bar{x} + \bar{t}\bar{d}),$$

then $\qquad \beta > 0 .$

Since $\qquad x^k + \bar{t}\, d^k \to \bar{x} + \bar{t}\bar{d} ,$

there exists K such that

$$f(x^k + \bar{t}d^k) + \frac{\beta}{2} < f(\bar{x}) \tag{3.11}$$

for all $k > K$, $k \in \mathcal{K}$.

However, by

$$f(x^{k+1}) = \min_{0 \le t \le \tau} f(x^k + td^k) \le f(x^k + \bar{t}d^k) ,$$

and

$$f(\bar{x}) \le f(x^{k+1})$$

for all k, we get

$$f(\bar{x}) \le f(x^k + \bar{t}d^k)$$

which contradicts (3.11).

In consequence

$$\bar{d} = 0 .$$

Considering (2.2) and (2.3), we deduce that $\bar{x}$ satisfies definition 1.1, and it is a stationary point of problem 2.1. □

Theorem 3.1. Any accumulation point $\bar{x}$ of any sequence generated by the algorithm is a Kuhn-Tucker point of problem 2.1.

Proof. As $\bar{x}$ is a stationary point of the problem, it is only necessary to prove that the Lagrange multipliers at $\bar{x}$ are positive. Note that they coincide with λ_{0i} and λ_i when $\bar{d}_0 = 0$.

Consider a sequence $\{x_k\}$, $k \in K$, converging to $\bar{x}$, and a constraint $g_h(x)$ so that

$$g_h(\bar{x}) = 0 \ .$$

As the method is strictly feasible

$$g_h(x^k) < 0 \quad \text{for all} \quad k.$$

In consequence, we can define a sequence

$$K' \subset K$$

so that
$$g_h(x^{k+1}) > g_h(x^k) \tag{3.12}$$

where
$$k \in K'.$$

Comparing (3.12) with (2.7), we deduce that

$$\gamma_h^k < 1 \quad ; \quad k \in K' \ .$$

Considering the rule of determination of γ_i, in step 3 of the algorithm, we get

$$\lambda_h^k \geq 0 \quad ; \quad k \in K' \ .$$

Then $\{\lambda_h^k\}$, $k \in K'$, will have an accumulation point $\bar{\lambda}_h$ verifying

$$\bar{\lambda}_h \geq 0 \ ,$$

and, as λ_h is a continuous function of x and Ω_a is compact,

$$\lambda_h(\bar{x}) = \bar{\lambda}_h \geq 0 .$$

In consequence, $\bar{x}$ is a Kuhn-Tucker point of the problem. □

4. EQUALITY CONSTRAINTS

In this section we consider the general non linear programming problem 1.1. A theory will be given in order to extend the domain of application of the algorithm stated in section 2 to problems with equality constraints.

This could be done in different ways. The simplest idea is to define a suitable penalty function of the equalities, and to minimize that function submitted only on the inequality constraints. Unfortunately, this approach brings up the numerical problems given by penalty functions.

Our approach consists in the establishment of an auxiliary optimization problem with inequality constraints, whose solution by means of the algorithm previously stated, gives the Kuhn-Tucker points of the original problem.

Mayne and Polack, in ref. [17], developed a similar idea. Our proposition is more restricted, because we state an auxiliary problem intended only for be solved by means of the method given before. We shall also consider the behaviour of the iterative process; a monotonous approximation to the equality constraints will be imposed.

Let us consider problem 1.1, and define the function

$$\theta_c(x) = f(x) - \sum_{i=m+1}^{m+p} c_i g_i(x) \tag{4.1}$$

where $c \equiv (c_{m+1},\ldots,c_{m+p})$ is constant.

The auxiliary problem is stated as

$$\min \ \theta_c(x) \tag{4.2}$$

$$\text{subject to} \quad g_i(x) \le 0 \quad ; \quad i = 1,\ldots,m+p. \tag{4.3}$$

Notice that even if θ_c is not continue, the sequence $\{x^k\}$ given by the application of the algorithm to this problem, does not traverse the discontinuity of θ_c.

Problem (4.2) must verify the same hypothesis that problem 2.1, for a given c. Then, we assume that exists a real number a, so that the region

$$\Omega_a \equiv \{x \in R^n \; ; \; g_i(x) \leq 0 \quad , \quad i=1,\dots,m+p, \; \theta_c(x) \leq a\}$$

is compact with non empty interior. In order to guarantee the existence of solutions of problem 1.1, it is necessary to add the following hypothesis :

Let $$\Omega^e \equiv \{x \in R^n \; ; \; g_i(x) = 0 \quad , \quad i = m+1,\dots,m+p\} ,$$

then $\Omega_a \cap \Omega^e$ is not empty.

We suppose also that f is c^1 and g_i are c^2 in Ω_a, and that all $x \in \Omega_a$ are regular points of the problem 1.1.

In consequence, θ_c is c^1 and all $x \in \Omega_a$ are regular points of problem (4.2).

We denote by

$$g^i \equiv (g_1,\dots,g_m)^T \quad , \quad g^e \equiv (g_{m+1},\dots,g_{m+p})^T ,$$

$$r^{ik} \equiv (r^k_1,\dots,r^k_m)^T \quad , \quad r^{ek} \equiv (r^k_{m+1},\dots,r^k_{m+p})^T ,$$

and by G^i, G^e, R^{ik}, R^{ek} the associated diagonal matrices.

<u>Theorem 4.1.</u> For any initial point x^0 verifying (4.3), there exists

$$c_i < \infty \quad ; \; i = m+1,\dots,m+p ,$$

such that any convergent sequence given by the algorithm stated in section 2 applied to the resolution of problem 4.2, converges to a Kuhn-Tucker point of problem 1.1.

Proof. Let us perform the Step 1 of the algorithm, applied to problem 4.2, in a point x^k. We obtain the linear system

$$d_0 = -[\nabla f + A^i\lambda_0 + A^e(\mu_0 - c)] \tag{4.4}$$

$$A^{iT} d_0 = -R^i G^i \lambda_0 \tag{4.5}$$

$$A^{eT} d_0 = -R^e G^e \mu_0 \quad , \tag{4.6}$$

where $\mu_0 \equiv (\mu_{m+1},\ldots,\mu_{m+p})^T$ is the vector of Lagrange multipliers corresponding to the equality constraints, $A^i \equiv \nabla g^i$, and $A^e \equiv \nabla g^e$.

We want to find c such that

$$\mu_{0i}^k > \theta > 0 \quad ; \quad i = m+1,\ldots,m+p \tag{4.7}$$

in all the points given by the algorithm. If $\{x^k\} \to x^*$, x^* is a Kuhn-Tucker point of 4.2, and if (4.7) is verified, then

$$\mu_{0i}(x^*) > 0 \quad ; \quad i = m+1,\ldots,m+p \ ,$$

and

$$g_i(x^*) = 0 \quad ; \quad i = m+1,\ldots,m+p.$$

In consequence, x^* will be a Kuhn-Tucker point of problem (4.2).

Substituting (4.4) in (4.5), we get

$$\lambda_0 = -(A^{iT}A^i - R^iG^i)^{-1} [A^{iT}\nabla f + A^{iT}A^e(\mu_0 - c)] \ , \tag{4.8}$$

and substitution of (4.4) and (4.8) in (4.6) gives

$$\mu = [A^{eT}(I-Z^i)A^e - R^eG^e]^{-1} [-A^{eT}(I-Z^i)\nabla f + A^{eT}(I-Z^i)A^e c] \tag{4.9}$$

where

$$Z^i = A^i(A^{iT}A^i - R^iG^i)^{-1} A^{iT} \ .$$

Note that, in consequence of lemma 3.1,

$$W^i = A^{iT}A^i - R^iG^i$$

is positive definite.

By lemma 3.2, $(I-Z^i)$ is positive semi definite, then

$$y^T A^{eT}(I-Z^i)A^e y \geq 0 \ .$$

Suppose now that $y \in R^p$ is such that

$$y^T A^{eT}(I-Z^i) A^e y = 0 \ ,$$

in consequence of lemma 3.3 we have that

$$v = A^e y$$

is contained in the subspace generated by the gradients of the active inequality constraints of problem 1.1. Because of the regularity assumption, we deduce that

$$y \equiv 0 \ .$$

In consequence, $A^{eT}(I-Z^i)A^e$ is positive definite, and also $[A^{eT}(I-Z)A^e-R^eG^e]$.

By means of equation (4.9) it is possible to find c, in each iteration, such that μ_0^k verifies condition (4.7). But we think that in general, and with the hypothesis that we have, it is not possible to find a unique value of c for all the iterates.

Suppose now that

$$r_i^k = 1/\mu_{0i}^k \ , \tag{4.10}$$

it follows from (4.9) that

$$\mu = c - [A^{eT}(I-Z^i)A^{e\ -1}\ \ A^{eT}(I-Z^i)\nabla f]. \tag{4.11}$$

In consequence, taking

$$c_i > \theta + \max_{x \in \Omega_a} \{e_i^T [A^{eT}(I-Z^i)A^e]^{-1}\ \ A^{eT}(I-Z^i)\nabla f\}\ , \tag{4.12}$$

when e_i is the i-th cartesian unitary vector, (4.7) is verified in all iterates. Since $A^{eT}(I-Z^i)A^e$ is positive definite and ∇f, ∇g_i are bounded in Ω_a, there is a finite c verifying (4.12).

Note that for such a value of c, r_i stated in (4.10) are positive in Ω_a and depend continuously of x. Then r_i^k are properly defined. □

In the algorithm for solving problem 1.1, we shall take r_i^k defined in (4.10). If we call

$$\lambda_{0i} = \mu_{0i} - c_i \quad ; \quad i = m+1,\dots,m+p \quad , \tag{4.13}$$

condition (4.12) becomes

$$c_i > \theta + \max_{x \in \Omega_a} \ (-\lambda_{0i}) \ .$$

Note that $\lambda_{0i}(x^*)$ are the Lagrange multipliers for the original problem 1.1.

The iterative algorithm for solving the general non linear programming problem 1.1 is stated as follows :

Let $\rho_0 > 0$, $\alpha \in (0,1)$, $\gamma_0 \in (0,1)$, $_i \geq 0$; $i = m+1,\dots,m+p$, and $r_i(x) > 0$; $i = 1,\dots,m$ continuous in Ω_a.

Step 0 : Select a strictly feasible point for the inequality constraints, $x^0 \in \Omega$, and the values of α, γ_0, ρ_0 and C_i ; $i = m+1,\dots,m+p$. If it is necessary, redefine the inequality constraints in a way that

$$g_i(x^0) \leq 0 \quad ; \quad i = m+1,\dots,m+p.$$

Set $\rho = \rho_0$.

Step 1 : Compute $\lambda_0 \in R^m$ and $d_0 \in R^n$ by solving the linear system of equations

$$d_0 = -[\nabla f(x) + \sum_{i=1}^{m+p} \lambda_{0i} \nabla g_i(x)]$$

$$d_0^T \nabla g_i(x) = - r_i(x) \lambda_{0i} g_i(x) \quad ; \quad i = 1,\ldots,m$$

$$d_0^T \nabla g_i(x) = - g_i(x) \quad ; \quad i = m+1,\ldots,m+p \tag{4.14}$$

If $d_0 = 0$, Stop.

Step 2 : If $c_i < -1.2\,\lambda_{0i}$, set $c_i = -2\lambda_{0i}$; $i = m+1,\ldots,m+p$.

Compute $Z = \sum_{i=1}^{m} \lambda_{0i} + \sum_{i=m+1}^{m+p} (\lambda_{0i} + c_i)$

and $\rho_1 = (1-\alpha)Z$.

If $0 < \rho_1 < \rho$, set

$\rho = \frac{1}{2}\rho_1$. (4.15)

Compute $\lambda \in R^m$ and $d \in R^n$ by solving the linear system of equations

$$d = - [\nabla f(x) + \sum_{i=1}^{m+p} \lambda_i \nabla g_i(x)]$$

$$d^T \nabla g_i = -[r_i(x)\lambda_i g_i(x) + \rho |d_0|^2] \quad ; \quad i = 1,\ldots,m$$

$$d^T \nabla g_i = -[\frac{\lambda_i + C_i}{\lambda_{0i} + C_i} g_i(x) + \rho |d_0|^2] \quad ; \quad i = m+1,\ldots,m+p \quad (4.16)$$

Setp 3 : Set $\gamma_i = \gamma_0$ if $\lambda_i \geq 0$ or $\gamma_i = 1$ if $\lambda_i < 0$ for the inequality constraints, and $\gamma_i = 0$ for the equalities.

Call τ the smallest τ_i such that

$$g_i(x + td) \leq \gamma_i g_i(x)$$

for $t \in (0,\tau_i)$ and $i = 1,m+p$.

Find $\bar{t}$ so that

$$\theta_c(x + \bar{t}d) = \min \theta_c(x + td)$$
$$\text{over} \quad t \in [0,\tau]$$

Step 4 : Set the new iterate

$$\bar{x} = x + \bar{t}d$$

Step 5 : Go to step 1.

Condition (4.15) was obtained by substituting (4.13) in (2.4). In step 3, γ_i was taken zero for the equality constraints, in order to allow them to be active in the next point.

In the numerical applications, we shall take

$$d^T \nabla g_i = - [g_i(x) + \rho \, |d_0|^2] \quad ; \quad i = m+1,\ldots,m+p \,, \tag{4.17}$$

in the place of (4.16). In this way, the numerical computations in step 2 are simplified.

With this modification all the theoretical development is valid, but the definition of Z given in (4.15) must be changed to

$$Z = \sum_{i=1}^{m} \lambda_{0i} + \sum_{i=m+1}^{m+p} (\lambda_{0i} + c_i) - g^{eT}(A^TA-R^iG^i)^{-1}e \tag{4.18}$$

In a way similar to lemma 3.4, it can be proved that in this case, d is a descent direction of θ_c.

5. NUMERICAL RESULTS

In general, in real applications, it is not possible to perform the exact minimization of the function included in step 3 of the algorithm. Instead we take $\bar{t}$ verifying a criterium defined by Wolf in ref. [26] and developed by Lemarechal in ref. [27].

The given algorithm has been applied to several test problems. We report here our experience with six problems, described in a work by Hock et al. [14]. We shall identify them with the same number as in the mentioned work.

Problem 35 - (Beale's problem) has 3 design variables and 4 linear equality constraints.

Problem 43 - (Rosen - Suzuki, [7]) has 4 design variables and 3 nonlinear inequality constraints.

Problem 78 - [3,18] has 5 design variables and 3 nonlinear equality constraints.

Problem 80 - (Powell, [21]) is a modification of problem 78; it has 5 design variables, 3 nonlinear inequality constraints and 10 linear inequality constraints.

Problem 86 - (Colville N° 1, [8]) has 5 design variables and 15 linear inequality constraints.

Problem 117 - (Colville N° 2, [8]) has 15 design variables, 5 nonlinear inequality constraints and 15 linear inequality constraints.

In all of them, the initial point is feasible for the inequality constraints and non feasible for the equalities. The iterative process was stopped with a value of the function correct to five significant digits, the inequalities verified, and the equalities verified with an error less than $10.^{-5}$.

The tests were performed on a HB-68 DPS/Multics computer. All the calculations were carried out in single precision (27 bit mantissa), except problem 117, calculated in double precision.

In Table 5.1 we give our final results and also intermediate results in which the objective function value is correct up to two significative digits.

Problem	Iterations	Func. and grad. evaluations	Function value
35	5	6	0.1123447
	9	11	0.1111125
43	5	9	-43.81453
	13	18	-43.99907
78	5	5	-2.959694
	12	12	-2.919709
80	3	4	0.05478925
	15	18	0.05394989
86	6	6	-32.03453
	9	9	-32.34851
117	34	38	32.81567
	49	64	32.34897

Table 5.1

Even if the purpose of the work of Hock et al. was not to study the efficiency of the tested nonlinear programming methods, it is very convenient to compare our results with those that they obtained with six different methods. In their work, the best performances are given by VFØ2AD and OPRQP programs. VFØ2AD was developed by Powell; it is an implementation of Wilson, Han and Powell's method [11,12,19,20].

OPRQP was developed by Biggs, based on his own method described in refs. [5,6]. Note that VFØ2AD solves a quadratic programming subproblem at each iteration, and OPRQP needs an active set strategy. Both programs approximate the Hessian of the Lagrangean of the problem, by means of a quasi-Newton method.

In the numerical tests shown in [14], in general VFØ2AD needed a less number of functions and gradient evaluations than OPRPQ; but in counterpart, OPRQP used less calculation time.

In the examples considered here, the number of evaluations with our method, generally goes between the number of VFØZAD and OPRQP. We estimate that computation time per iteration used by our method is similar to that used by Bigg's approach.

Table 5.1 shows that the final convergence of the present method is slow. It seems that this may be improved with the used of quasi-Newton technique

Considering that the present is a feasible method and that it doesn't make use of quasi-Newton techniques, we conclude that the numerical results are very satisfactory.

The method proved also to be very reliable. This is due to the fact that active set strategies are unnecessary, and that the linear search scheme doesn't introduce discontinuities.

Acknowledgement :

The author would like to thank Dr. Claude Lemaréchal for the extense discussions and the very helpfull suggestions.

REFERENCES

[1] J. Abadie, J. Carpentier, The generalization of the Wolfe reduced gradient method to the case of nonlinear constraints, in : Optimization, R. Fletcher ed., Academic Press, 1969.

[2] J. Abadie, Méthode du gradient réduit généralisé : Le code GRGA, Note HI 1756/00, Electricité de France, Paris, 1975.

[3] D.P. Bertsekas, On penalty and multiplier methods for constrained minimization, SIAM J. Control and Optimization, Vol. 14, n° 2, 2/1978.

[4] M.C. Biggs, Constrained minimization using recursive equality quadratic programming, in: Numerical methods for non linear optimization, F.A. Lootsma ed., Academic Press, London, New York, 1971.

[5] M.C. Biggs, On the convergence of some constrained minimization algorithms based on recursive quadratic programming, Journal of the Institute of Mathematics and its applications, Vol. 21, n° 1 (1978), 67-82.

[6] M.C. Bartholomew-Biggs, An improved implementation of the recursive quadratic programming method for constrained minimization, Technical Report N° 105, Numerical Optimization Centre, The Hatfield Polytechnic, Hatfield, England, 1979.

[7] C. Charambolus, Nonlinear least path optimization and nonlinear programming, Mathematical Programming, Vol. 12 (1977), 195-225.

[8] R.A. Colville, A comparative study on nonlinear programming codes, IBM Scientific Center, Report 320-2949, New York, 1968.

[9] R. Fletcher, Methods for nonlinear constraints, NATO Advanced Research Symposium, Cambridge, 7/1981.

[10] D. Gabay, Reduced quasi-Newton methods with feasibility improvement for nonlinear constrained optimization, 10th International Symposium on Mathematical Programming, Montreal, 10/1979.

[11] S.P. Han, Superlinearly convergent variable metric algorithms for general nonlinear programming problems, Mathematical Programming 11(1976), 263-282.

[12] S.P. Han, A globally convergent method for nonlinear programming, J.O.T.A., Vol. 22, n° 3, 7/1977.

[13] J. Herskovits and N. Zouain, A nonlinear programming algorithm for structur optimization problems, 10th International Symposium on Mathematical Programming, Montreal, 10/1979.

[14] W. Hock and K. Schittkowski, Test examples for nonlinear programming codes, Lecture Notes in Economics and Methematical Systems, Springer Verlag, 1981.

[15] C. Lemarechal, A view of line searches, in: Optimization and Optimal Control Editors Auslender, Oettli and J. Stoer, Lectures Notes in Control and Information Sciences, 30, Springer Verlag, 1981, 59-78.

[16] D.G. Luenberger, Introduction to linear and nonlinear programming, Addison-Wesley P.C., 1973.

[17] D.Q. Mayne and E. Polack, Feasible direction algorithms for optimization problems with equality and inequality constraints, Mathematical Programming 11 (1976) 67-80.

[18] M.J.D. Powell, A method for nonlinear constraints in minimization problems, in: Optimization, R. Fletcher ed., Academic Press, New York, 1969.

[19] M.J.D. Powell, A fast algorithm for nonlinearly constrained optimization calculations, in: Proceedings of the 1977 Dundee Conference on Numerical Analysis, Lectures Notes in Mathematics, Springer Verlag, Berlin, Heidelber New York, 1978.

[20] M.J.D. Powell, The convergence of variable metric methods for nonlinearly constrained optimization calculations, in: Nonlinear programming 3, O.L. Mangasarian, R.R. Meyer, S.M. Robinson eds., Academic Press, New York, San Francisco, London, 1978.

[21] M.J.D. Powell, Algorithms for non linear constraints that use Lagrangian functions, Mathematical Programming, 14 (1978) 224-248.

[22] R.W.H. Sargent, Reduced-gradient and projection methods for non linear programming, in Numerical Methods for Constrained Optimization, edited by Gill P.E. and W. Murray, Academic Press, London, 1974.

[23] S. Sagenreich, N. Zouain, J. Herskovits, An optimality criteria method based on slack variables concept for large structural optimization, Symp. on Applications of Computer Meth. in Engineering, Los Angeles, USA, (1977).

[24] R.A. Tapia, Diagonalized multiplier methods and quasi-Newton methods for constrained optimization", J.O.T.A., vol. 22, n° 2 6/1977.

[25] D.M. Topkins and A.R. Veinott Jr., On the convergence of some feasible direction algorithms for nonlinear programming, J. SIAM Control, Vol. 5, n° 2, 1967.

[26] P. Wolfe, Convergence conditions for ascent methods, SIAM Review 11 (1969) 226-235.